

SON-2047

PATENT APPLICATION

#2  
C. Kananen

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

In re the Patent Application of

Akihiki KOH et al.

Serial No. To be assigned

Filed: March 12, 2001

For: DATA PROCESSING APPARATUS

)  
)  
)ATT: APPLICATION BRANCH  
)  
)  
)  
)  
)

11000 U.S. PTO  
09/802857  
03/12/07

CLAIM TO PRIORITY UNDER 35 USC 119

Commissioner for Patents  
Washington, D.C. 20231

Sir:

The benefit of the filing date of the following prior application filed in the following foreign country is hereby requested and the right of priority provided under 35 U.S.C. 119 is hereby claimed:

Japanese Patent Appl. No. 2000-076331 filed March 14, 2000

In support of this claim, filed herewith is a certified copy of said original foreign application.

Respectfully submitted,

Date: March 12, 2001

*B. P.* *47,255*  
for Ronald P. Kananen  
Registration No. 24,104

**RADER, FISHMAN & GRAUER, PLLC**  
Lion Building  
1233 20<sup>th</sup> Street, N.W.  
Washington, D.C. 20036  
Tel: (202) 955-37650  
Customer No. 23353

## 日 本 国 特 許 庁

PATENT OFFICE  
JAPANESE GOVERNMENT

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日

Date of Application:

2000年 3月14日

出 願 番 号

Application Number:

特願2000-076331

出 願 人

Applicant (s):

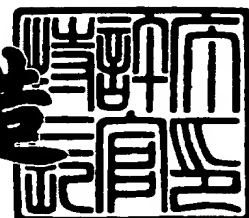
ソニー株式会社



2000年12月 1日

特許庁長官  
Commissioner,  
Patent Office

及 川 耕 造



出証番号 出証特2000-3100784

【書類名】 特許願

【整理番号】 0000009402

【提出日】 平成12年 3月14日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 9/06

【発明者】

【住所又は居所】 神奈川県横浜市保土ヶ谷区神戸町 1 3 4 番地 ソニー・  
エルエスアイ・デザイン株式会社内

【氏名】 黄 明彦

【発明者】

【住所又は居所】 神奈川県横浜市保土ヶ谷区神戸町 1 3 4 番地 ソニー・  
エルエスアイ・デザイン株式会社内

【氏名】 三瓶 勉

【発明者】

【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
内

【氏名】 渡辺 信久

【発明者】

【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社  
内

【氏名】 菊地 章浩

【特許出願人】

【識別番号】 000002185

【氏名又は名称】 ソニー株式会社

【代表者】 出井 伸之

【代理人】

【識別番号】 100094053

【弁理士】

【氏名又は名称】 佐藤 隆久

【手数料の表示】

【予納台帳番号】 014890

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【包括委任状番号】 9707389

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 データ処理装置

【特許請求の範囲】

【請求項 1】

プログラムが格納されているメモリから読み出した命令コードに従って所定のデータ処理を行うプログラム実行手段を含むデータ処理装置であって、

上記メモリに格納されているプログラムにおいてバグが含まれる部分の先頭を示すバグアドレスを保持するアドレス保持手段と、

データ処理を行うとき、上記メモリからプログラムの読み出しを行うプログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとを比較し、アドレスが一致した場合、一致信号を出力する比較手段と

を有し、

上記プログラム実行手段は、上記比較手段によって上記一致信号が出力されたとき、現在実行している命令を中断し、予め設定されたアドレステーブルによって指し示されたプログラムアドレスから命令コードを読み出し、当該読み出した命令コードに従って処理を行う

データ処理装置。

【請求項 2】

上記比較手段は、上記プログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとが一致したとき、上記一致信号として割り込み要求信号を出力する割り込み要求手段

を有する請求項 1 記載のデータ処理装置。

【請求項 3】

初期処理によって、外部から入力されたバグ修正用プログラムを格納する書き込み可能なメモリと、

上記バグ修正用プログラムが格納されているメモリ領域の先頭アドレスを格納する割り込みベクトルと

有する請求項 2 記載のデータ処理装置。

【請求項 4】

上記プログラム実行手段は、上記割り込み要求信号を受けたとき、現在実行されている命令を中断し、上記割り込みベクトルによって指し示されたアドレスから上記バグ修正用プログラムを読み込み、それに従って処理を行う割り込み処理手段

を有する請求項 3 記載のデータ処理装置。

【請求項 5】

上記割り込み処理手段は、上記バグ修正用プログラムの実行を終了したとき、上記バグ修正用プログラムの最後に格納されているアドレスに従って、割り込み処理を終了した後中断したプログラムへの戻りアドレスを設定する

請求項 4 記載のデータ処理装置。

【請求項 6】

プログラムが格納されているメモリから読み出した命令コードに従って所定のデータ処理を行うプログラム実行手段を含むデータ処理装置であって、

上記メモリに格納されているプログラムにおいてバグが含まれる部分の先頭を示すバグアドレスを保持するアドレス保持手段と、

データ処理を行うとき、上記メモリからプログラムの読み出しを行うプログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとを比較し、アドレスが一致した場合、一致信号を出力する比較手段と

を有する基本単位を上記プログラムに含まれているバグの数分設けられ、

上記プログラム実行手段は、上記複数の比較手段のうち、何れかの比較手段から上記一致信号が出力されたとき、現在実行している命令を中断し、予め設定されたアドレステーブルによって指し示されたプログラムアドレスから命令コードを読み出し、当該読み出した命令コードに従って処理を行う

データ処理装置。

【請求項 7】

上記比較手段は、上記プログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとが一致したとき、上記一致信号として割り込み要求信号を出力する割り込み要求手段

を有する請求項 6 記載のデータ処理装置。

【請求項 8】

初期処理によって、外部から入力された複数のバグ修正用プログラムを格納する書き込み可能なメモリと、

上記バグ修正用プログラムが格納されているメモリ領域の先頭アドレスを格納する割り込みベクトルと

有する請求項 7 記載のデータ処理装置。

【請求項 9】

上記プログラム実行手段は、上記割り込み要求信号を受けたとき、現在実行されている命令を中断し、上記割り込みベクトルによって指し示されたアドレスから上記バグ修正用プログラムを読み込み、それに従って処理を行う割り込み処理手段

を有する請求項 8 記載のデータ処理装置。

【請求項 10】

上記割り込み処理手段は、割り込みが発生した回数を記録する割り込み回数記録手段と、

上記割り込み回数記録手段に記録されている割り込み回数に応じて、上記メモリに格納されている複数のバグ修正用プログラムのうち、所定のバグ修正用プログラムに分岐する分岐手段と

を有する請求項 9 記載のデータ処理装置。

【請求項 11】

上記バグ修正用プログラムの最後に当該バグ修正用プログラムが終了したあと、元のプログラムに戻るときの戻り先のアドレスが格納され、上記割り込み処理手段は、何れかのバグ修正用プログラムの実行を終了したあと、当該バグ修正用プログラムの最後に格納されている上記戻りアドレスに応じて、元のプログラムへ戻るときの戻りアドレスを設定する

請求項 9 記載のデータ処理装置。

【請求項 12】

上記割り込み回数記録手段は、上記バグ修正用プログラムを格納する書き込み

可能なメモリのうち所定の番地のメモリであり、当該メモリの内容は、上記割り込み処理手段によって書き換えられる

請求項10記載のデータ処理装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、データ処理装置、例えば、少なくともCPU（中央処理装置）とCPUにプログラムを供給するメモリを含み、製品化の後メモリに組み込まれているプログラムのバグを修正できるデータ処理装置に関するものである。

【0002】

【従来の技術】

一般的なデータ処理装置、例えば、マイクロコンピュータ（Micro-Computer）は、演算、制御機能を持つCPUとプログラムを格納するメモリなどが一つの半導体チップ上に集積されている。データ処理を行うとき、CPUは、プログラムカウンタによって指示されたアドレスに応じて上記メモリから命令コードを順次読み出して、命令コードによって指示された動作を行うので、プログラムによって予め設定した手順で所定の演算または処理を行うことができる。

【0003】

CPUのプログラムを格納するメモリとして、通常、読み出し専用メモリ（ROM：Read Only Memory）が使われている。プログラムは、製造段階でROMに書き込まれ、製造後にプログラムの書き換えはできない。

【0004】

ところが、プログラムは応用機器の開発と同時に製作され、応用機器が未完成な状態でプログラムが製作されるものが多い。また、応用機器が製品化のあと、仕様の変更などによって、それまでに製作され、すでにマイクロコンピュータに組み込まれた制御プログラムの一部分には応用機器との整合性が取れなくなることもある。さらに、大規模なシステムでは、プログラムの容量も膨大なものになり、製品化前のテストのみではすべての不具合を発見することが困難であるため、製品化の後にプログラムのバグ（誤り）が発見されることが多い。



## 【 0 0 0 5 】

マイクロコンピュータを製品化した後にバグを発見した場合、プログラムを修正することができないため、すでに製造したマイクロコンピュータは使用できず、無駄になってしまい、損失を生じることがある。また、バグを修正したプログラムを製作しても、新たにマイクロコンピュータを発注して製造するために、コストと時間の両面に不利が生じることがある。

## 【 0 0 0 6 】

このような欠点を解消するために、マイクロコンピュータに予めバグを修正する機能を組み込むことが提案されていた。即ち、従来のマイクロコンピュータにバグを発見した場合、バグが含まれるプログラムを実行せずに、代わりにバグが修正されたプログラムを実行することによって、製品化した後でもバグの修正を可能にする。以下、このようなバグ修正機能を有するマイクロコンピュータの一例を示し、その構成及び動作を説明する。

## 【 0 0 0 7 】

図 6 は、バグを修正可能なマイクロコンピュータの一構成例を示すブロック図である。図示のように、このマイクロコンピュータは、CPU 1 0、バグ修正回路 2 0、ROM 3 0、シリアルインターフェース（S I O：Serial Input Output）4 0、ランダムアクセスメモリ（RAM：Random Access Memory）5 0 及びバス（B u s）6 0 によって構成されている。

## 【 0 0 0 8 】

CPU 1 0 は、ROM 3 0 から読み出されたプログラムに従って、所定の演算処理または制御を行う。

バグ修正回路 2 0 は、図 7 に示すように、バグアドレス設定レジスタ 2 2、一致検出回路 2 4、分岐命令発生回路 2 6 及び選択回路 2 8 によって構成されている。マイクロコンピュータが初期化のとき、S I O 4 0 を介して外部メモリ 7 0 からプログラムのバグ部分の先頭アドレス（以下、これをバグアドレスという）が読み出され、バグアドレス設定レジスタ 2 2 に設定される。プログラムが実行されるとき、一致検出回路 2 4 によって、バグアドレスとアドレスバス A D R B U S から入力されるプログラムアドレスとが比較され、比較結果に応じた選択制

御信号  $S_C$  が発生され、選択回路 28 に出力される。

【0009】

分岐命令発生回路 26 は、絶対分岐命令コードを発生する。CPU10 がこの絶対分岐命令を実行することによって、プログラムカウンタのカウント値が絶対分岐命令によって指示された分岐先アドレスに設定され、次にこのアドレスに格納されたプログラムコードが読み出され、CPU10 によって実行される。

選択回路 28 は、分岐命令発生回路 26 によって発生された分岐命令コードまたは ROM30 から読み出されたプログラムコードの何れかを選択して、データバス DATBUS に出力する。一致検出回路 24 によって、プログラムアドレスとバグアドレスが一致しないと判断された場合、選択制御信号  $S_C$  が論理“0”、例えば、ローレベルに保持される。これに応じて、選択回路 28 は ROM30 から読み出されたプログラムコードを選択して、データバスに出力する。一方、一致検出回路 24 によって、プログラムアドレスとバグアドレスが一致したと判断された場合、選択制御信号  $S_C$  が論理“1”、例えば、ハイレベルに保持される。これに応じて、選択回路 28 は分岐命令発生回路 26 によって発生された分岐命令を選択して、データバスに出力する。

【0010】

即ち、通常動作時に、CPU10 はプログラムアドレスに応じて、ROM30 からプログラムコードを読み出して、それに従って処理を行う。プログラムアドレスが予め設定されたバグアドレスに達したとき、バグ修正回路 20 によって分岐命令を CPU10 に供給することによって、バグプログラムが実行されず、分岐命令で示したアドレスに格納されているバグ修正後のプログラムが実行され、バグの回避を実現する。

【0011】

ROM30 には、CPU10 のプログラム及び処理用データなどが格納されている。なお、これらのプログラム及びデータが予め製作され、製造時に ROM30 に組み込まれる。

SIO40 は、マイクロコンピュータと外部に設けられている記憶手段、例えば、外部メモリ 70 との間にデータの伝達を行うためのシリアル通信手段である

。マイクロコンピュータが初期化のとき、S I O 4 0 を介して外部メモリ 7 0 から、バグを含むプログラムの先頭アドレス、末尾アドレス及びバグを含むプログラムを置き換える新しいプログラムを読み込み、読み込まれたアドレス及びプログラムをそれぞれ所定のメモリ、例えば、R A M 5 0 に格納する。

R A M 5 0 は、マイクロコンピュータが初期化のとき、S I O 4 0 を介して外部メモリ 7 0 から読み込まれたアドレス及びプログラムを格納する。

#### 【 0 0 1 2 】

通常、図 6 に示すようにマイクロコンピュータチップ以外に、プログラム及び処理用データを格納するメモリ 7 0 が設けられている。メモリ 7 0 には、C P U 1 0 の初期化プログラム、バグを修正した後のプログラム、さらに、処理用データ、例えば、マイクロコンピュータを初期化するための初期化データ、及び特定のデータ処理を行うためのパラメータなどが格納されている。

#### 【 0 0 1 3 】

以下、図 6 に示すデータ処理装置の動作について説明する。

初期化のとき、S I O 4 0 を介して外部メモリ 7 0 から種々のデータが読み込まれる。例えば、R O M 3 0 に格納されているプログラムにバグが含まれる部分の先頭アドレス（バグアドレス）、末尾アドレス、さらにバグを修正した後のプログラムなどがそれぞれ読み込まれる。バグアドレスがバグ修正回路 2 0 にあるバグアドレス設定レジスタ 2 2 に書き込まれ、バグを修正した後のプログラムは R A M 5 0 の所定の領域に書き込まれる。バグ修正回路 2 0 の分岐命令発生回路 2 6 は、R A M 5 0 にバグ修正後のプログラムが格納される領域の先頭に分岐する分岐命令コードを発生する。

#### 【 0 0 1 4 】

初期化の後、C P U 1 0 は、プログラムカウンタによって発生したプログラムアドレスに応じて、R O M 3 0 からプログラムコードを順次読み出して実行する。そして、バグ修正回路 2 0 において、プログラムアドレスとバグアドレス設定レジスタ 2 2 に設定されたバグアドレスとが比較され、プログラムアドレスがバグアドレスに達するまでに、一致検出回路 2 4 から論理“0”の選択制御信号 S<sub>C</sub> が出力されるので、R O M 3 0 から読み出されたプログラムコードが選択回路

28によって選択され、データバスに出力され、CPU10によって、データバスから入力されたプログラムコードが読み込まれ、実行される。

【0015】

プログラムアドレスがバグアドレス設定レジスタ22に設定されたバグアドレスと一致したとき、バグ修正回路20において一致検出回路24から出力される選択制御信号 $S_C$ が論理“1”になる。これに応じて、選択回路28によって、分岐命令発生回路26によって発生された分岐命令及び分岐先のコードが選択され、データバスに出力される。CPU10によって、データバスから分岐命令及び分岐先のコードが入力され、実行されるので、プログラムカウンタのカウント値が分岐先のアドレス、例えば、RAM50にバグ修正後のプログラムが格納されている領域の先頭アドレスに設定される。これに応じて、次の動作周期からRAM50に格納されているバグ修正後のプログラムのコードが順次データバスに出力され、CPU10によって読み込まれ、実行される。バグ修正後のプログラムの末尾に、ROM30に格納されているバグが含まれるプログラム部分の最後のアドレスの次のアドレスに分岐する絶対分岐命令が書き込まれているので、CPU10がこの分岐命令を実行することによって、プログラムカウンタのカウント値がROM30のバグある部分の最後のアドレスの次のアドレスに書き換えられ、次の動作周期からこのアドレスからプログラムコードが順次読み出され、CPU10によって実行される。

【0016】

上述した動作によって、ROM30に格納されているプログラムのうち、バグがある部分が回避され、バグ修正後のプログラムが実行される。そして、バグ修正後のプログラムが実行されたあと、ROM30にあるバグ部分の次のメモリアドレスに戻り、処理が続行することができる。

【0017】

【発明が解決しようとする課題】

ところで、上述した従来のマイクロコンピュータにおいて、バグ修正回路には、バグアドレス設定レジスタ、一致検出回路のほか、分岐命令発生回路及び選択回路がそれぞれ組み込まれているので、回路構成が複雑になり、バグ修正回路を

組み込むことによって回路規模が増加する。特に、ROMに格納されているプログラムに複数のバグが存在した場合、それぞれのバグを回避するために、バグ設定レジスタ、一致検出回路、分岐命令発生回路及び選択回路からなる基本単位を複数組設ける必要があり、バグ修正回路がさらに大規模なものになってしまうという不利益がある。

【0018】

また、マイクロコンピュータにプログラムを格納するメモリとして、ROMの代わりに電氣的に書き換え可能なフラッシュメモリを用いる方法も提案されているが、フラッシュメモリを用いたマイクロコンピュータは一般的にチップ単価がROMを用いたものより高くなり、さらにフラッシュメモリは製造時にプログラムの組み込みができず、製品が完成したあとプログラムの書き込み作業によってチップごとにプログラムが書き込まれるので、製品化まで時間がかかり、コストの増加を招き、量産化には不向きである。

【0019】

本発明は、かかる事情に鑑みてなされたものであり、その目的は、回路規模を大幅に増大させることなく、製造コストの増加を抑制しながら、バグを含むプログラムを実行せず、代わりにバグ修正後のプログラムを実行することによって組み込みプログラムのバグを回避できるデータ処理装置を提供することにある。

【0020】

【課題を解決するための手段】

上記目的を達成するため、本発明のデータ処理装置は、プログラムが格納されているメモリから読み出した命令コードに従って所定のデータ処理を行うプログラム実行手段を含むデータ処理装置であって、上記メモリに格納されているプログラムにおいてバグが含まれる部分の先頭を示すバグアドレスを保持するアドレス保持手段と、データ処理を行うとき、上記メモリからプログラムの読み出しを行うプログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとを比較し、アドレスが一致した場合、一致信号を出力する比較手段とを有し、上記プログラム実行手段は、上記比較手段によって上記一致信号が出力されたとき、現在実行している命令を中断し、予め設定されたアドレステーブルによ

て指し示されたプログラムアドレスから命令コードを読み出し、当該読み出した命令コードに従って処理を行う。

## 【 0 0 2 1 】

また、本発明のデータ処理装置では、好適には、上記比較手段は、上記プログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとが一致したとき、上記一致信号として割り込み要求信号を出力する割り込み要求手段を有する。

## 【 0 0 2 2 】

また、本発明では、好適には、初期処理によって、外部から入力されたバグ修正用プログラムを格納する書き込み可能なメモリと、上記バグ修正用プログラムが格納されているメモリ領域の先頭アドレスを格納する割り込みベクトルとを有する。

## 【 0 0 2 3 】

また、本発明では、好適には、上記プログラム実行手段は、上記割り込み要求信号を受けたとき、現在実行されている命令を中断し、上記割り込みベクトルによって指し示されたアドレスから上記バグ修正用プログラムを読み込み、それに従って処理を行う割り込み処理手段を有する。

## 【 0 0 2 4 】

また、本発明では、好適には、上記割り込み処理手段は、上記バグ修正用プログラムの実行を終了したとき、上記バグ修正用プログラムの最後に格納されているアドレスに従って、割り込み処理を終了した後中断したプログラムへの戻りアドレスを設定する。

## 【 0 0 2 5 】

また、本発明のデータ処理装置は、プログラムが格納されているメモリから読み出した命令コードに従って所定のデータ処理を行うプログラム実行手段を含むデータ処理装置であって、上記メモリに格納されているプログラムにおいてバグが含まれる部分の先頭を示すバグアドレスを保持するアドレス保持手段と、データ処理を行うとき、上記メモリからプログラムの読み出しを行うプログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとを比較し、アド

レスが一致した場合、一致信号を出力する比較手段とを有する基本単位を上記プログラムに含まれているバグの数分設けられ、上記プログラム実行手段は、上記複数の比較手段のうち、何れかの比較手段から上記一致信号が出力されたとき、現在実行している命令を中断し、予め設定されたアドレステーブルによって指し示されたプログラムアドレスから命令コードを読み出し、当該読み出した命令コードに従って処理を行う。

## 【 0 0 2 6 】

また、本発明では、好適には、上記比較手段は、上記プログラムアドレスと上記アドレス保持手段に保持されているバグアドレスとが一致したとき、上記一致信号として割り込み要求信号を出力する割り込み要求手段を有する。

## 【 0 0 2 7 】

また、本発明では、好適には、初期処理によって、外部から入力された複数のバグ修正用プログラムを格納する書き込み可能なメモリと、上記バグ修正用プログラムが格納されているメモリ領域の先頭アドレスを格納する割り込みベクトルとを有する。

## 【 0 0 2 8 】

また、本発明では、好適には、上記割り込み処理手段は、割り込みが発生した回数を記録する割り込み回数記録手段と、上記割り込み回数記録手段に記録されている割り込み回数に応じて、上記メモリに格納されている複数のバグ修正用プログラムのうち、所定のバグ修正用プログラムに分岐する分岐手段とを有する。

## 【 0 0 2 9 】

さらに、本発明では、上記バグ修正用プログラムの最後に当該バグ修正用プログラムが終了したあと、元のプログラムに戻るときの戻り先のアドレスが格納され、上記割り込み処理手段は、何れかのバグ修正用プログラムの実行を終了したあと、当該バグ修正用プログラムの最後に格納されている上記戻りアドレスに応じて、元のプログラムへ戻るときの戻りアドレスを設定する。

## 【 0 0 3 0 】

本発明によれば、データ処理装置に、バグアドレスを保持する保持手段、例えば、バグアドレス設定レジスタ及びバグアドレスとプログラムアドレスとを比較

する比較手段が設けられ、プログラムにバグが発見された場合、バグの部分の先頭アドレスがバグアドレス保持手段に格納され、プログラムが実行されるとき、比較手段によってプログラムアドレスと保持手段に保持されているバグアドレスが比較される。比較の結果、プログラムアドレスとバグアドレスが一致したとき一致信号が発生され、これに応じてプログラム実行手段、例えば、CPUは現在実行されている命令を中断し、予め設定されたアドレステーブルによって指し示されたプログラムアドレスから、バグが修正されたあとのプログラムを読み出して実行するので、バグのあるプログラムが回避される。そして、バグが修正されたプログラムの最後に、もとのプログラムへの戻り番地が設定されるので、プログラム実行手段は、この戻り番地に分岐することによって、プログラムの中バグのある部分の次のプログラムに従って処理を続行することができる。

【0031】

【発明の実施の形態】

#### 第1実施形態

図1は本発明に係るデータ処理装置の第1の実施形態を示す回路図である。

図示のように、本実施形態のデータ処理装置は、CPU10、メモリ（ROM）30、シリアルインターフェース（SIO）40、メモリ（RAM）50、バス（データバスとアドレスバスを含む）60及びバグ修正回路100によって構成されている。通常、これらの部分回路及びバスが一つの半導体チップに集積され、いわゆるワンチップマイクロコンピュータが構成される。半導体チップの外に、外部メモリ70が設けられている。

【0032】

以下、それぞれの部分回路の構成について説明する。

CPU10は、図示しないプログラムカウンタによって出力されるプログラムアドレスに応じて、ROM30から読み出された命令コードを読み出し、それに応じた演算または他の処理を行う。さらに、CPU10は、外部から入力された割り込み信号 $S_A$ に応じて、所定の割り込み処理を行う割り込み処理機能を有する。



## 【 0 0 3 3 】

本発明では、CPU 10の割り込み処理機能を利用して、バグプログラムの処理を行う。なお、CPU 10の割り込み処理機能は、通常のプログラム実行時に外部から入力される要求信号に応答して、即時に処理を行う場合に利用される。通常CPU 10が処理可能な複数の割り込みがそれぞれ外部の割り込み要求に割り当てられている。このため、通常の処理に利用されない割り込み処理、例えば、マイクロコンピュータのテスト用に設けられている中断割り込みを利用することで、通常の割り込み処理に影響を与えることなく、バグ修正後のプログラムを実行できる。

## 【 0 0 3 4 】

ROM 30は、CPU 10のプログラム及び処理用データなどを記憶する。ROM 30の記憶データは、製造時にROM 30に組み込まれ、その後読み出しのみができ、書き換えはできない。

## 【 0 0 3 5 】

SIO 40は、マイクロコンピュータと外部に設けられている記憶手段、例えば、外部メモリ 70との間にデータ伝達を行うためのシリアル通信手段である。通常、マイクロコンピュータが初期化のとき、SIO 40を介して外部メモリ 70からROM 30に格納されているプログラムの中にバグを含むプログラムの先頭アドレス、末尾アドレス及びバグが修正されたプログラムを読み込み、読み込まれたアドレス及びプログラムをそれぞれ所定の部分、例えば、RAM 50に格納する。

RAM 50は、マイクロコンピュータが初期化のとき、SIO 40を介して外部メモリ 70から読み込まれたアドレス及びバグを修正した後のプログラムを格納する。

## 【 0 0 3 6 】

バグ修正回路 100は、予め設定されたバグアドレスとCPU 10が実行されるプログラムアドレスとを比較し、当該比較結果に応じてCPU 10に割り込み要求信号  $S_A$  を発生する。以下、図 2 を参照しつつ、バグ修正回路 100の構成について詳細に説明する。

## 【 0 0 3 7 】

図示に示すように、バグ修正回路 1 0 0 は、バグアドレス設定レジスタ 1 1 0 及び一致検出回路 1 2 0 によって構成されている。初期化のとき、上述したように、外部メモリ 7 0 から S I O 4 0 を介してバグを含むプログラムの先頭アドレス（以下、これをバグアドレスと表記する）が読み込まれる。バグアドレスがデータバス D A T B U S に出力され、データバスを介してバグアドレス設定レジスタ 1 1 0 に書き込まれる。

C P U 1 0 がプログラムを実行するとき、プログラムカウンタのカウント値がプログラムアドレスとして、アドレスバス A D R B U S に出力される。一致検出回路 1 2 0 は、アドレスバスから入力されるプログラムアドレスとバグアドレス設定レジスタ 1 1 0 に設定されたバグアドレスとを比較し、当該比較結果に応じて割り込み要求信号  $S_A$  を発生する。例えば、プログラムアドレスとバグアドレスが一致しないとき、割り込み要求信号  $S_A$  がハイレベルに保持され、プログラムアドレスとバグアドレスが一致したとき、割り込み要求信号  $S_A$  がローレベルに保持される。

## 【 0 0 3 8 】

C P U 1 0 は、一致検出回路 1 2 0 から出力される割り込み要求信号  $S_A$  を受けて、割り込み処理を行う。例えば、割り込み要求信号  $S_A$  の立ち下がりエッジにおいて、C P U 1 0 に割り込み要求が発生され、C P U 1 0 は、現在実行中の命令コードの動作周期が終了したとき割り込み処理を行う。即ち、一致検出回路 1 2 0 によって、プログラムアドレスと予め設定されているバグアドレスとが一致したことが検出され、これに応じて割り込み要求信号  $S_A$  がローレベルに保持される。C P U 1 0 は、割り込み信号  $S_A$  の立ち下がりエッジにおいて割り込みに応答して、現在実行されているプログラムを中断し、割り込み処理を行う。

## 【 0 0 3 9 】

図 3 は、本実施形態のデータ処理装置におけるメモリの内容を示すメモリ配置図であり、図 4 は、本実施形態のデータ処理装置の動作を示すフローチャートである。以下、図 3 及び図 4 を参照しながら、本実施形態のデータ処理装置の動作について説明する。

## 【 0 0 4 0 】

データ処理装置が動作を開始したあと、まず、ステップ S 1 に示すように初期化処理が行われる。このとき、データ処理装置は S I O 4 0 を介して、外部メモリ 7 0 に格納されている初期化プログラム（イニシャルプログラム）を読み出して R A M 5 0 の所定のメモリ領域に展開する。また、この初期化処理において、外部メモリ 7 0 から R O M 3 0 に格納されているプログラムの中に、バグのあるプログラムの部分の先頭アドレス、即ち、バグアドレスが読み出され、バグ修正回路 1 0 0 のバグアドレス設定レジスタ 1 1 0 に設定される。

## 【 0 0 4 1 】

初期化処理によって、例えば、R A M 5 0 の所定のメモリ領域に、初期化データが格納される。図 3 に示すように、例えば、R A M 5 0 の 0 0 0 0 H 番地からリセットベクトル（R E S E T V E C T）、割り込みベクトル（I N T V E C T）及び中断処理ベクトル（A B O R T V E C T）がそれぞれ格納される。なお、この中断処理ベクトルは、バグ修正回路 1 0 0 の一致検出回路 1 2 0 からローレベルの割り込み要求信号  $S_A$  が出力されたとき、C P U 1 0 の分岐先を示すアドレスである。

## 【 0 0 4 2 】

初期化処理が行われたあと、R O M 3 0 に格納されているプログラムが順次読み出され、実行される。例えば、図 3 に示すように、番地 0 1 0 0 H から以降のプログラムは R O M 3 0 に格納されているプログラムである。このプログラムのなか、バグアドレス B A D R 0 から末尾アドレス B A D R 1 までのプログラムにバグが存在する場合、上述した初期化処理によって、バグアドレスとして、B A D R 0 がバグ修正回路 1 0 0 のバグアドレス設定レジスタ 1 1 0 に書き込まれる。

## 【 0 0 4 3 】

アドレスバスに出力されたプログラムアドレスが更新されるたびに、バグ修正回路 1 0 0 において、プログラムアドレスとバグアドレスとの比較が行われる（ステップ S 3）。プログラムアドレスとバグアドレスが一致しない場合、割り込み要求信号  $S_A$  がハイレベルのままに保持され、C P U 1 0 は通常の処理を行う。

。即ち、プログラムカウンタによって指し示したROM30のアドレスから次のプログラムコードを読み出し、それに従って所定の処理を行う。

【0044】

プログラムアドレスとバグアドレスが一致した場合、即ち、図3に示すメモリ配置図において、プログラムアドレスがバグが含まれるプログラム部分の先頭のアドレスBADR0に達したとき、バグ修正回路100の一致検出回路120からローレベルの割り込み要求信号 $S_A$ が出力される（ステップS4）。これを受けて、CPU10において割り込みが発生し、図3に示すように、中断ベクトル（ABORTVECT）に格納されているベクトル（例えば、F000H）がプログラムカウンタに設定される（ステップS5）。

【0045】

そして、次の動作周期からCPU10は割り込み処理ルーチンを実行する。図3の例では、割り込み処理ルーチンは、中断ベクトルによって指し示したF000H番地からのメモリ領域に格納されている。なお、このメモリ領域は、RAM50にある領域であり、ここに格納されているバグ修正後のプログラムは、上述したように初期化処理のとき、SIO40を介して外部メモリ70から読み込み、RAM50に格納されたものである。

【0046】

割り込み処理において、CPU10は、アドレスバスに出力されるプログラムアドレスによって示したメモリ番地から順次命令コードを読み出して実行する。即ち、図3に示すように、RAM50においてメモリ番地F000Hからの領域に格納されているバグ修正後のプログラムが順次実行される（ステップS6）。そして、このプログラムの最後に、割り込み終了後の戻り番地を示す命令コード（例えば、図3の例では、“RET BADR+1”）が格納されている。この命令コードを読み込んだCPU10は、割り込み処理を終了させ、そして、プログラムカウンタに戻り番地である“BADR1+1”を設定する（ステップS7）。

【0047】

図3に示すように、BADR1は、ROM30に格納されているプログラムの

うち、バグが存在するプログラム部分の最後のメモリ番地である。このため、CPU 10に戻り番地として、“B A D R 1 + 1”が設定されることによって、割り込み処理が終了したあと、CPU 10は、バグが存在するプログラム部分の次のプログラムコードを読み込み、実行を続行する。

## 【 0 0 4 8 】

上述したように、プログラムアドレスが予め設定されたバグアドレスと一致した場合、CPU 10が割り込み処理を実行し、そして、割り込み処理が終了したあと、バグが存在するプログラム部分の次のメモリ番地からプログラムコードを読み込み処理の続行することによって、ROM 30に格納されているプログラムのうち、バグの存在する部分が実行されることなく、代わりに割り込み処理ルーチンとしてRAM 5に格納されているバグ修正後のプログラムが実行されるので、バグのあるプログラムが回避される。

## 【 0 0 4 9 】

本実施形態において、マイクロコンピュータなどほとんどすべての処理装置に備え付けの割り込み処理機能を利用して、プログラムのバグの部分を回避することができる。通常、マイクロコンピュータは優先権レベルの異なる複数の割り込みを処理でき、これらの割り込み処理のうち適切なものを選択して利用することによって、通常の処理にほとんど影響を与えることなくバグのあるプログラム部分を回避可能である。例えば、種々のマイクロコンピュータの中に、テスト時に現在実行中のプログラムを一次中断させ、その後実行を再開させるために設けられている中断（A B O R T）割り込み処理機能を備えたものがある。このようなマイクロコンピュータにおいて、テスト以外のとき、この中断処理機能がほとんど使用されていないため、これを利用することによって、図3に示すように中断ベクトル（A B O R T V E C T）としてバグ修正後のプログラムの格納領域の先頭のアドレスを格納することによって、プログラムの実行中、プログラムアドレスがバグアドレスに達したとき、中断割り込みが発生し、RAM 50に格納されている割り込み処理ルーチン、即ちバグ修正後のプログラムが実行され、バグのあるプログラム部分が回避される。

## 【 0 0 5 0 】

このため、本実施形態のデータ処理装置において、バグ修正回路 1 0 0 の構成は従来に比べて簡略化され、バグアドレスを格納するバグアドレス設定レジスタ 1 1 0 とアドレスを比較する一致検出回路 1 2 0 のみによって構成できる。さらに、従来のバグ修正回路のように、プログラムアドレスが選択回路などを通すことなく、選択回路のゲート遅延によって生じた動作遅延を回避できる。

## 【 0 0 5 1 】

さらに、本実施形態のデータ処理装置において、ROM 3 0 のプログラムに複数のバグがある場合、バグ修正回路 1 0 0 において、修正するバグの数分だけバグアドレス設定レジスタ 1 1 0 と一致検出回路 1 2 0 からなるユニットを設けることによって対処できる。次に、複数のバグを処理できる本発明のデータ処理装置の第 2 の実施形態について説明する。

## 【 0 0 5 2 】

第 2 実施形態

図 5 は本発明に係るデータ処理装置の第 2 の実施形態を示す回路図である。

本発明の第 2 の実施形態のデータ処理装置は、バグ修正回路を除けば、他の各部分は第 1 の実施形態のデータ処理装置とほぼ同じである。このため、図 5 は本実施形態のデータ処理装置におけるバグ修正回路 1 0 0 a の構成のみを示している。

## 【 0 0 5 3 】

ここで、例えば、ROM 3 0 のプログラムの中に二つのバグが発見され、これらのバグに対してそれぞれバグ修正後のプログラムが製作され、初期化処理によって RAM 5 0 にロードされる。

## 【 0 0 5 4 】

図 5 に示すように、バグ修正回路 1 0 0 a は、バグアドレス設定レジスタ 1 1 0 - 1, 1 1 0 - 2 及び一致検出回路 1 2 0 - 1, 1 2 0 - 2 によって構成されている。初期化処理によって、バグアドレス設定レジスタ 1 1 0 - 1 には、一目のバグアドレス B A D R 0 - 1 が格納され、バグアドレス設定レジスタ 1 1 0 - 2 には、二目のバグアドレス B A D R 0 - 2 が格納されている。

## 【 0 0 5 5 】

一致検出回路 1 2 0 - 1 と 1 2 0 - 2 は、それぞれ二つのバグアドレスとプログラムアドレスとを比較し、アドレスが一致したとき、ローレベルの信号  $S_{A1}$  と  $S_{A2}$  をそれぞれ出力する。

CPU 1 0 の割り込み処理に余裕がある場合、一致検出回路 1 2 0 - 1 と 1 2 0 - 2 の出力信号  $S_{A1}$  と  $S_{A2}$  をそれぞれ異なる割り込み要求信号として CPU 1 0 に入力し、これに応じて CPU 1 0 はそれぞれ異なる割り込み処理として受け付け、それぞれのバグ修正後のプログラムを実行し、二つのバグを修正することが可能である。しかし、一般的に CPU 1 0 の処理可能な割り込みの数が制限されており、複数のバグ処理を一つの割り込みに割り当てる必要がある。この場合、図 5 に示すように、各一致検出回路 1 2 0 - 1 と 1 2 0 - 2 の出力信号  $S_{A1}$  と  $S_{A2}$  が AND ゲート 1 3 0 に入力され、AND ゲート 1 3 0 の出力信号  $S_A$  を割り込み要求信号として CPU 1 0 に入力される。

## 【 0 0 5 6 】

CPU 1 0 がプログラム実行するとき、プログラムアドレスが何れかのバグアドレスと一致したとき、一致検出回路 1 2 0 - 1 または 1 2 0 - 2 の出力信号がローレベルとなり、AND ゲート 1 3 0 の出力信号  $S_A$  がローレベルとなる。これに応じて、CPU 1 0 に割り込み要求が発生され、CPU 1 0 は中断ベクトルによって指し示された割り込み処理ルーチンのプログラムを実行する。

## 【 0 0 5 7 】

ここで、複数のバグ修正プログラムが順序よく実行するために、割り込み処理ルーチンの先頭に、それぞれのバグ修正プログラムに分岐する分岐処理プログラムが設けられる。例えば、RAM 5 0 の所定のメモリ番地をカウンタレジスタとして指定し、初期化のとき当該カウンタレジスタをクリアし（0 に設定する）、割り込みルーチンが実行されるたびに当該カウンタレジスタの値に 1 を足す。CPU 1 0 はカウンタレジスタの値に応じて、何回目の割り込み、即ち、何番目のバグを修正するかを判断できるので、これに応じて正しいバグ修正プログラムに分岐し、プログラムコードを読み出し所定の処理を実行する。

## 【 0 0 5 8 】

以上説明した本発明の第2の実施形態によって、プログラムに複数のバグが発見された場合、バグの数に応じてバグ修正回路100aにバグアドレス設定レジスタと一致検出回路からなるユニットを設けて、複数の一致検出回路の出力信号に応じて、論理ゲートによって割り込み要求信号 $S_A$ を発生し、CPU10に入力する。CPU10は、割り込み要求を受けたとき、割り込みベクトルによって指し示した割り込み処理ルーチンからプログラムコードを読み出し実行する。割り込み処理ルーチンの先頭に、割り込み処理の回数をカウントするカウンタレジスタの値に応じて、何番目のバグを修正するかを判断でき、それに応じて複数のバグ修正後のプログラムのうち、正しいものに分岐する。

## 【 0 0 5 9 】

本実施形態において、バグ修正回路110aの基本単位であるバグアドレス設定レジスタと一致検出回路のユニットが単純な構成を有するので、複数のバグを修正する場合、複数のユニットが設けられてもバグ修正回路110aの構成が小規模に抑えられる。そして、複数のバグ修正処理は一つの割り込み処理ルーチンに割り当てることができ、複数のバグ修正後のプログラムへの分岐処理は、ソフトウェアによって実現できるので、ハードウェアの増加が必要最小限に抑えることが可能であり、また、割り込み処理ルーチンは初期化のとき外部メモリから、RAMにロードされるので、プログラムの変更を容易に実現でき、複数のバグが発見された場合、それぞれに対して柔軟に対処することができる。

## 【 0 0 6 0 】

本発明のデータ処理装置は、上述したようにプログラムに含まれるバグを修正することができる。なお、本発明は、プログラムに含まれるバグの修正のみならず、例えば、データ領域に含まれているバグを修正することも可能である。この場合、バグが存在するデータ、即ち誤りのあるデータのアドレスがバグアドレス設定レジスタに記憶され、CPUによってこの誤ったデータを読み出そうとするとき、一致検出回路に応じて割り込みが発生し、CPUは割り込み処理ルーチンにおいて正しいデータを取得することができる。



【 0 0 6 1 】

## 【発明の効果】

以上説明したように、本発明のデータ処理装置によれば、製品化のあとにROMに組み込まれたプログラムにバグが発見された場合、チップを作り直すことなく、バグの修正を行うことができる。データ処理装置の初期化において、バグ修正後のプログラムをRAMにロードし、プログラム実行時にプログラムアドレスとバグアドレスが一致したときに割り込みが発生され、CPUの割り込み処理機能を用いて、バグ修正後のプログラムを実行し、バグのあるプログラム部分を回避できるので、バグ修正回路のハードウェア構成を簡略化でき、バグ修正機能の追加によるコスト増を最小限に抑制可能である。

さらに、複数のバグを修正する場合、RAMにロードされている複数のバグ修正後のプログラムへの分岐はソフトウェアによって実現でき、ハードウェアの増加を必要最小限に抑えることができる利点がある。

## 【図面の簡単な説明】

## 【図 1】

本発明に係るデータ処理装置の第 1 の実施形態を示すブロック図である。

## 【図 2】

本実施形態のバグ修正回路の構成を示すブロック図である。

## 【図 3】

本実施形態のデータ処理装置におけるメモリの内容を示す図である。

## 【図 4】

本実施形態のデータ処理装置の動作を示すフローチャートである。

## 【図 5】

本発明に係るデータ処理装置の第 2 の実施形態を示すブロック図であり、バグ修正回路の構成を示すブロック図である。

## 【図 6】

従来のデータ処理装置の一例を示すブロック図である。

## 【図 7】

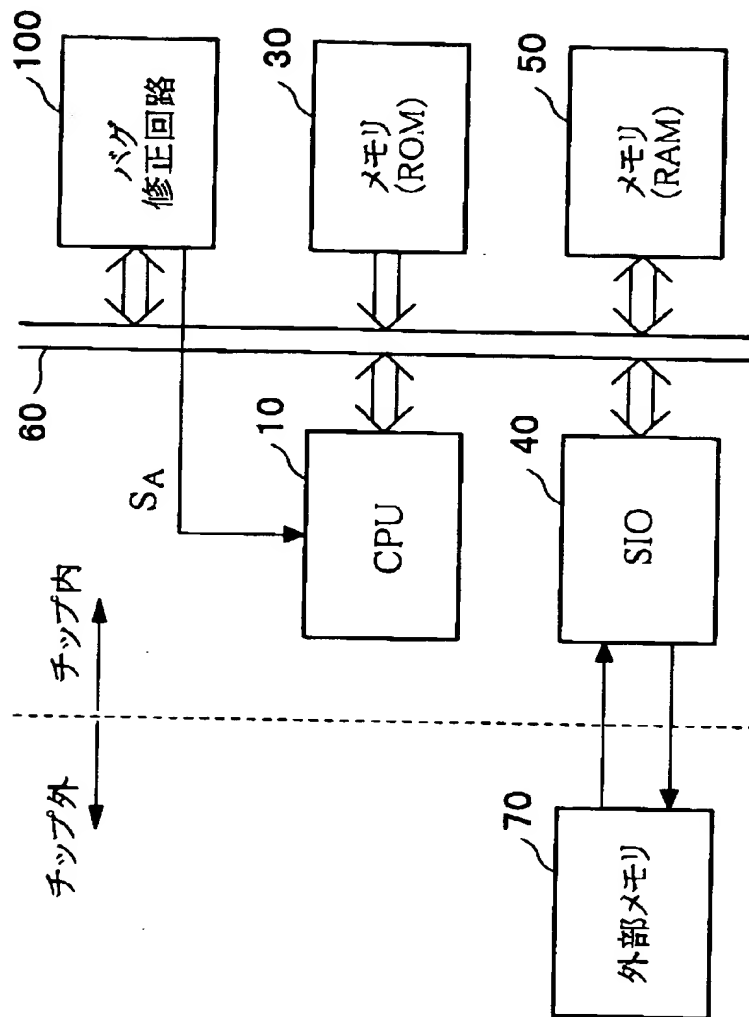
従来のデータ処理装置のバグ修正回路の構成を示すブロック図である。

【符号の説明】

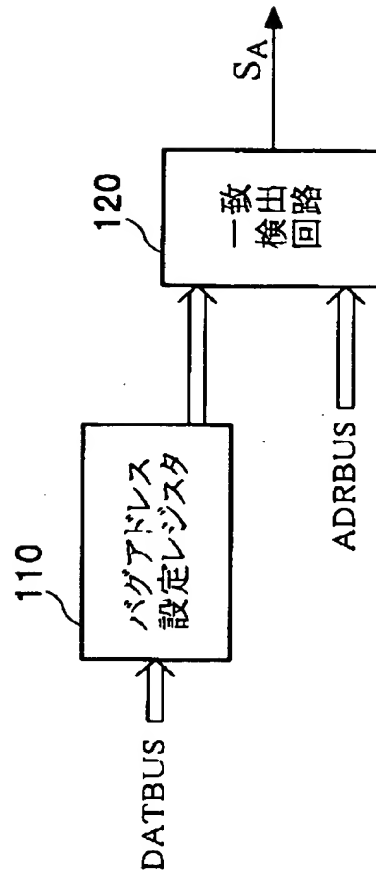
- 1 0 … C P U、
- 2 0 … バグ修正回路、
  - 2 2 … バグアドレス設定レジスタ、
  - 2 4 … 一致検出回路、
  - 2 6 … 分岐命令発生回路、
  - 2 8 … 選択回路、
- 3 0 … R O M、
- 4 0 … S I O、
- 5 0 … R A M、
- 6 0 … バス、
- 7 0 … 外部メモリ、
- 1 0 0 … バグ修正回路、
  - 1 1 0 … バグアドレス設定レジスタ、
  - 1 2 0 … 一致検出回路。

【書類名】 図面

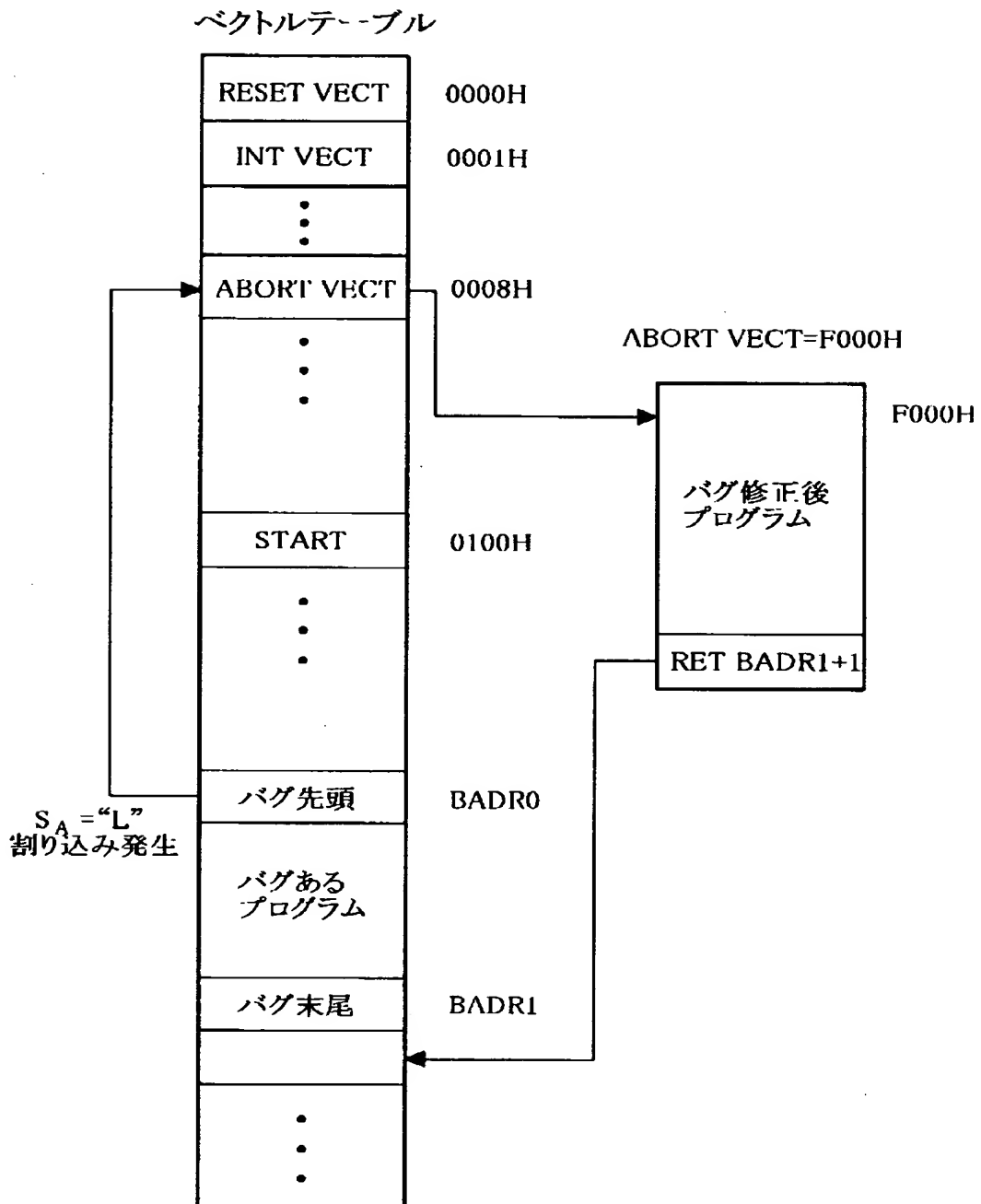
【図 1】



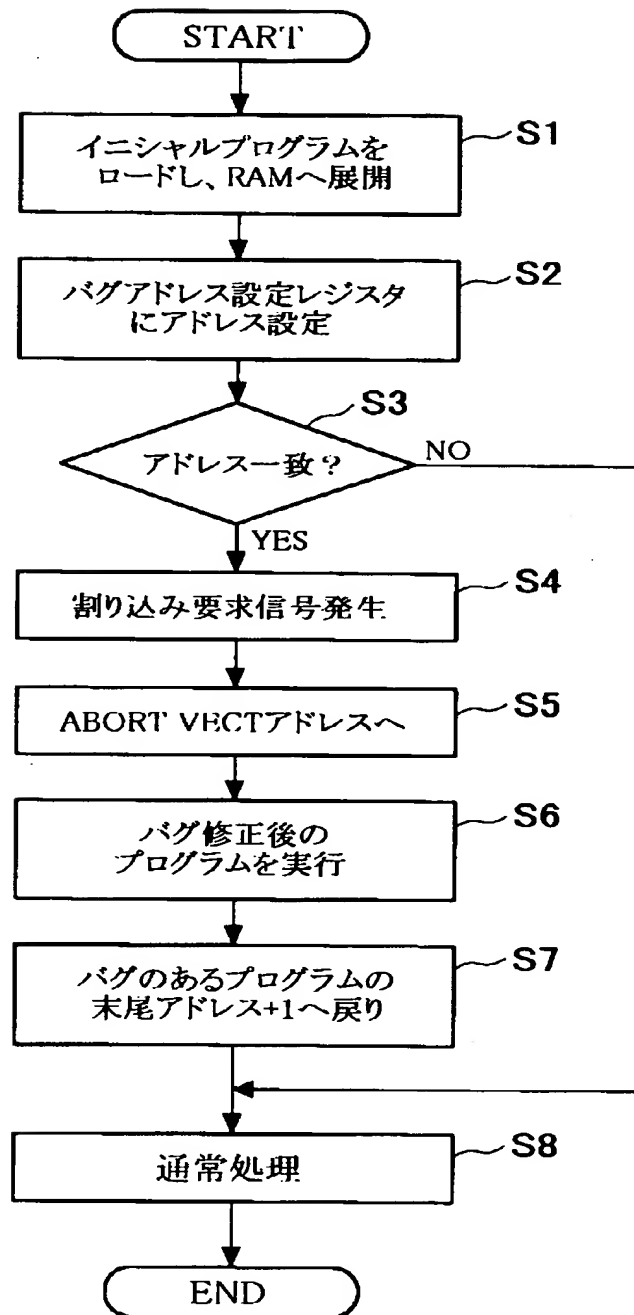
【図 2】



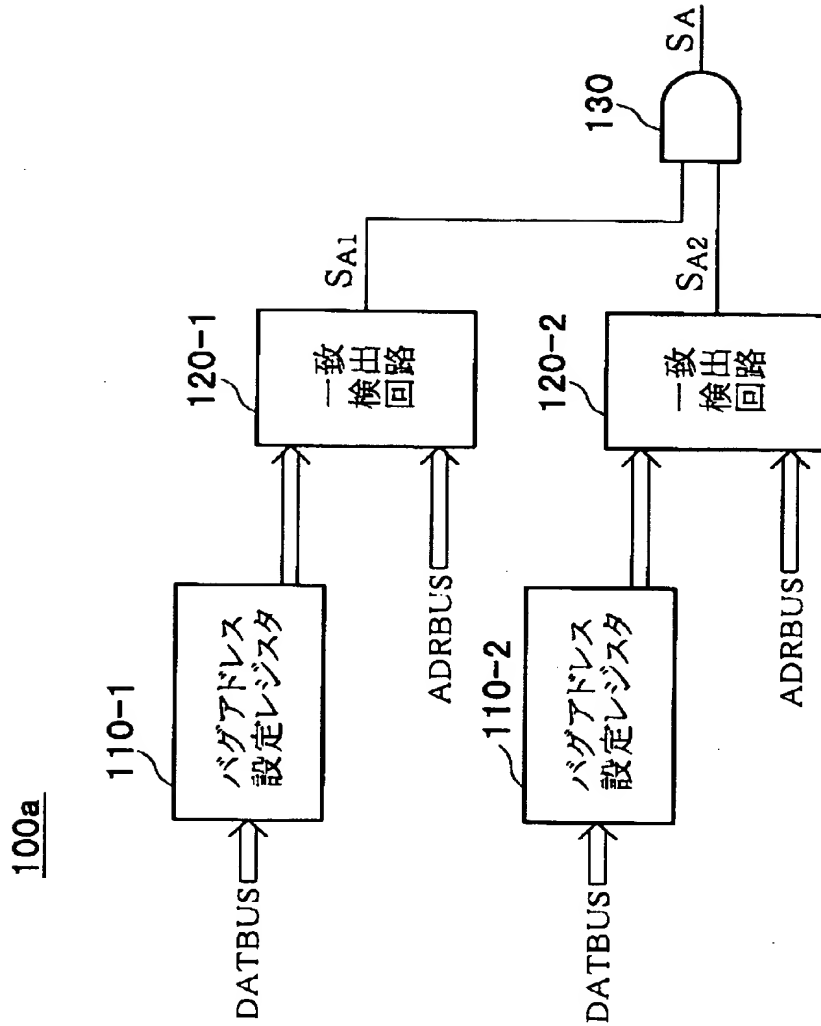
【図 3】



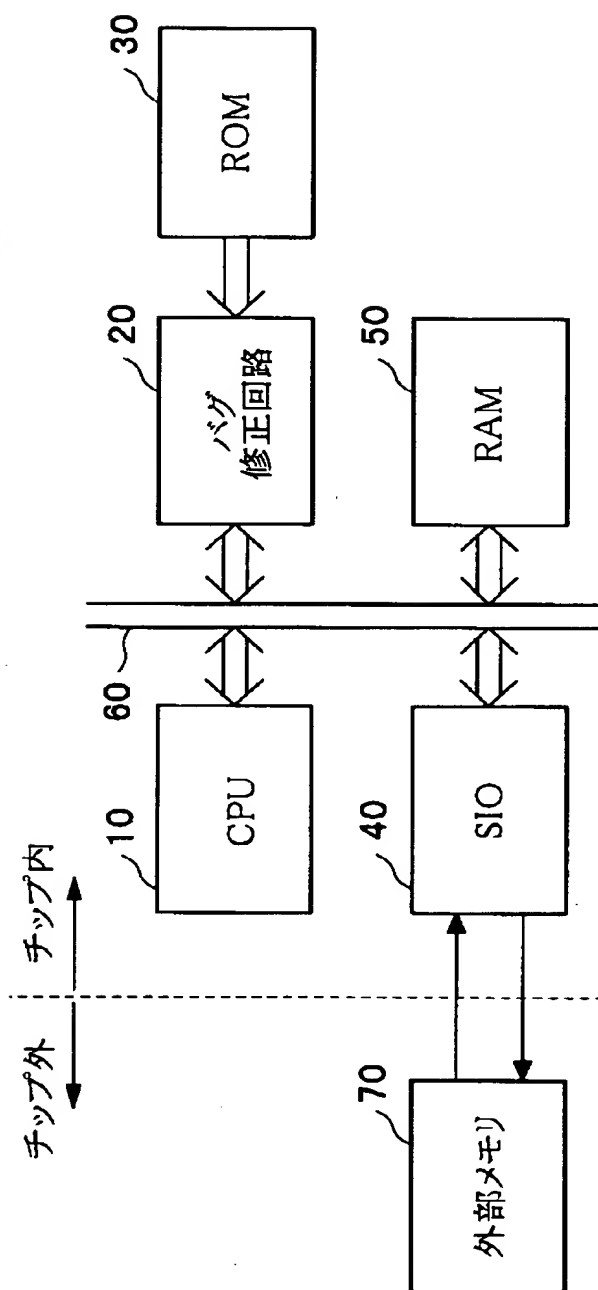
【図 4】



【図 5】

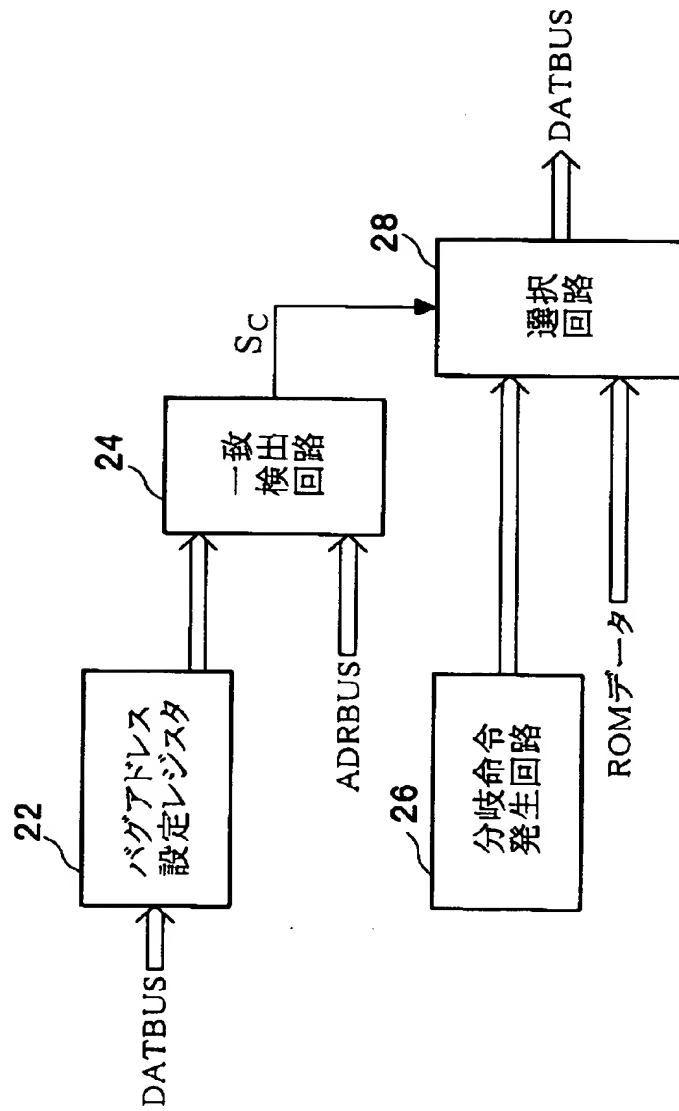


【図 6】





【図 7】



【書類名】 要約書

【要約】

【課題】 回路規模を増大させることなく、製造コストの増加を抑制しながら、バグを含むプログラム部分に代えてバグ修正後のプログラムを実行でき、組み込みプログラムのバグを回避できるデータ処理装置を提供する。

【解決手段】 プログラムアドレスが予め設定されたバグアドレスと一致した場合、バグ修正回路 1 0 0 から出力される割り込み要求信号に応じて C P U 1 0 が割り込み処理を行い、バグ修正後のプログラムを実行する。割り込み処理が終了したあと、バグが存在するプログラム部分の次のメモリ番地からプログラムコードを読み込み処理の続行することによって、R O M 3 0 に格納されているプログラムのうち、バグの存在する部分が実行されることなく、代わりに割り込み処理ルーチンとして R A M 5 に格納されているバグ修正後のプログラムが実行されるので、バグのあるプログラムが回避される。

【選択図】 図 1

出 願 人 履 歴 情 報

識別番号 [000002185]

1. 変更年月日 1990年 8月30日  
[変更理由] 新規登録  
住 所 東京都品川区北品川6丁目7番35号  
氏 名 ソニー株式会社